Contrôle de formation d'un réseau de drones à base d'apprentissage par renforcement

Nicola Roberto Zema[1] †, Mirwaisse Djanbaz[1], Dominique Quadri[1], Steven Martin[1], Enrico Natalizio[2] et Omar Shrit[1]

[1] *LRI - Université Paris-Saclay, Orsay, France* [2] *LORIA - Université de Lorraine, Vandœuvre lès Nancy, France*

Nous présentons une solution innovante basée sur un algorithme d'apprentissage par renforcement, le *Q-learning*, pour le contrôle de formation d'un réseau de drones par un unique opérateur. Pour suivre automatiquement le drone maître, le seul téléguidé, tous les autres n'utilisent que les puissances de signal reçues durant les communications ad hoc. Grâce à ces seules valeurs obtenues en temps-réel, la formation peut être parfaitement maintenue en appliquant notre schéma comportemental. Les expérimentations, menées sous ns-3, montrent l'efficacité de notre approche.

Mots-clefs : Drones, flotte de drones, contrôle de formation, réseau ad hoc, apprentissage par renforcement, Q-learning

1 Introduction

Après être longtemps resté limité au secteur militaire, le marché du drone a connu ces dernières années une croissance exponentielle avec l'avènement des drones civils (professionnels et de loisir). Ce phénomène a ouvert de nombreux axes de recherche, tant au niveau matériel que logiciel, et a permis l'apparition de nouvelles applications bouleversant différents secteurs d'activité (transports, bâtiment, agriculture, observation, surveillance, ...). Le pilotage est devenu aisé et accessible à tous, avec des calculateurs embarqués de plus en plus performants et miniaturisés, une autonomie accrue, une stabilité renforcée et des fonctions préprogrammées. De nouveaux modèles apparaissent régulièrement, rivalisant d'inventivité, allant de simples ballons à des engins tout-terrain capables de voler, rouler et naviguer, en passant par des multicoptères aux possibilités multiples. Mais un nouveau type d'utilisation se développe pour surmonter les limitations d'un unique drone, avec des applications encore à imaginer : les réseaux de drones. Ces flottes ont pour ambition de mener des missions difficiles ou trop coûteuses par des drones individuels. Par exemple, un groupe de drones peut observer, surveiller ou suivre des cibles spécifiques (personnes, véhicules, ...) dans de vastes zones. Des drones interconnectés peuvent également permettre une infrastructure réseau dans les airs afin d'offrir une couverture plus efficace que les réseaux de communication classiques (manifestations, événements sportifs, zones sinistrées, ...). Par ailleurs, les drones sont maintenant quasiment tous équipés d'une caméra 2D ou 3D. Mais un seul drone ne peut observer en temps-réel une région hostile/inaccessible ou derrière un obstacle, en raison de la perte de connexion avec le centre de commande. Certaines solutions existantes fournissent une connectivité via des réseaux d'infrastructure (4G, réseaux satellites). Cependant, cette connectivité est fortement contrainte par la couverture de l'opérateur.

Nous proposons ici une solution efficace pour permettre à un unique opérateur, avec une seule télécommande, de piloter un ensemble de drones volant en formation, dans des environnements intérieurs et extérieurs, pour assurer le service demandé. Pour ce faire, seul un drone est contrôlé à distance (le drone maître), les autres (les suiveurs) se déplacant automatiquement pour maintenir la formation initiale. La solution proposée ne requiert aucune infrastructure spécifique ni aucun matériel supplémentaire ou dédié, en particulier de module GPS dont les informations de positionnement peuvent être approximatives ou inexistantes (notamment dans des environnements intérieurs). Plus précisément, notre méthode, décrite dans la Section 2, permet de contrôler la formation d'un réseau de drones en adaptant un algorithme d'apprentissage par renforcement [Wat89], à savoir le *Q-learning*. Les drones suiveurs peuvent alors se (re)positionner

†Ce travail a été réalisé dans le cadre du projet Wizard, dont les fonds proviennent d'un programme Européen FEDER.

automatiquement en s'appuyant uniquement sur les puissances de signal reçues, ou RSSI (*Received Signal Strength Indication*), durant leurs communications ad hoc, selon le schéma comportemental défini. Bien que des modèles aient été proposés, par exemple pour gérer des essaims de drones [WQXC14] ou coordonner une flotte de drones pour accomplir des actions complexes [LB18], très peu de travaux mettent en oeuvre leur approche en situation réelle [CBF18]. Plusieurs travaux ont déjà étudié l'utilisation du RSSI pour maintenir la formation entre robots [KOK+10, WQXC14, GTK17]. Cependant, cette utilisation est retreinte au problème suivant : un robot donné doit localiser puis échanger les données de localisation et propager les informations au reste du groupe [ZTHK17]. De même, le Q-learning a été proposé pour traiter des problèmes de patrouilles de drones [Per08] et récemment pour contrôler une flotte de drones [HG17], mais sans prise en compte de la qualité de service demandée pour remplir la mission et en nécessitant un matériel spécialisé de type vidéo.

Notre approche, qui consiste à n'utiliser que les puissances de signal reçues pour contrôler la formation et à adapter l'algorithme d'apprentissage en conséquence est non seulement originale et efficace, mais permet également de garantir que la QoS (en termes de débit) soit toujours satisfaite. Plus précisément, les drones adaptent systématiquement et automatiquement leur position pour maintenir le même RSSI, garantissant ainsi le débit nécessaire à l'application pour fonctionner correctement. La solution proposée a été implantée sous forme protocolaire et testée sous ns-3. Les résultats sont présentés dans la Section 3.

2 RSSI et Q-learning pour le contrôle de formation

L'apprentissage par renforcement [Wat89] est une classe de méthodes ayant pour objectif d'apprendre à optimiser une récompense au cours du temps, à partir d'expériences [CK02]. Il considère un agent qui doit décider de façon automatique et autonome une action dans un univers incertain, modélisé par un Processus de Décision Markovien (MDP). Dans notre cadre d'étude, seul le drone maître est contrôlé par un opérateur, les drones suiveurs se déplaçant automatiquement en fonction du maître. Chaque drone suiveur (qui est un agent) doit prendre des décisions (actions) dans un environnement (ici la simulation) afin de maximiser une récompense. Plus la formation initiale est maintenue et plus l'agent est récompensé. Ainsi, le but de l'algorithme d'apprentissage par renforcement est d'apprendre la politique optimale, c'est-à-dire celle qui indique aux drones suiveurs dans quelle direction ils doivent se déplacer de telle sorte à rester le plus proche de la formation initiale. Plus précisément, l'environnement sera modélisé sous la forme d'un MDP composé d'un ensemble d'états finis S, d'un ensemble d'actions A et d'une fonction de récompense $Q : S \times A \rightarrow r$. Un état correspond à la situation de l'environnement perçue par le drone. En l'occurrence, les drones ont seulement comme information les différents RSSI courants des drones voisins, ainsi que l'historique des précédents RSSI et actions choisies.

Prenons l'exemple de trois drones dont la formation initiale est un triangle isocèle (voir Figure 1). Chaque drone suiveur nécessite de recevoir les informations des deux autres drones. En supposant que les puissances de signal reçues sont symétriques, un état est alors défini comme un ensemble composé par le triplet des puissances entre les drones à l'instant précédent, la dernière action que le drone a effectuée et le triplet des puissances mesurées juste après cette action.

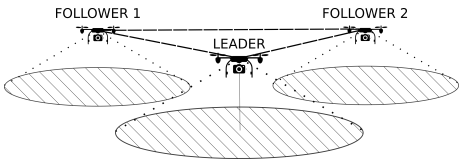

FIGURE 1: Trois drones en formation triangle

Afin de limiter l'espace des états, il est possible de discrétiser les puissances de signal en les arrondissant à l'entier le plus proche. Ainsi, un élément de S pourrait être par exemple : `{(-57dB, -58dB, -62dB), "aller à gauche", (-58dB, -57dB, -63dB)}`. Chaque transition donne lieu à une récompense (positive ou négative). L'algorithme va donc chercher à maximiser la somme des récompenses que le drone reçoit sur une période de temps donnée. La valeur de cette récompense est déterminée en comparant les positions des drones suiveurs obtenues à la suite d'actions et les positions attendues. Plus l'erreur est faible, plus la récompense est élevée. Enfin, nous définissons une fonction $Q : S \times A \rightarrow r$ permettant d'estimer la qualité d'un couple (état, action). Plus cette valeur est élevée, plus la récompense cumulée que nous pouvons espérer obtenir dans le futur en effectuant cette action à partir de cet état est élevée. Ainsi, pour

chaque état, la fonction donne une valeur aux quatre actions possibles (aller à gauche, à droite, avancer, reculer) dans notre contexte. Un *Q-tableau* est associé à cette fonction, comprenant les différents états (en lignes) et les quatre actions possibles pour un drone (en colonnes). La valeur d'une case correspond donc à la qualité de l'action pour l'état considéré. Au démarrage, toutes les valeurs sont nulles. A partir de l'état initial, l'action permettant à la fonction Q de retourner la plus grande valeur est choisie. Une fois cette action réalisée, une récompense est accordée. Cette dernière permet de mettre à jour la fonction Q selon la formule suivante : $Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha(r_t + \gamma \times \max_a Q(s_{t+1},a))$ où $Q(s_t,a_t)$ représente la récompense précédente, α le taux d'apprentissage, r_t la récompense, γ le facteur d'actualisation et $\max_a Q(s_{t+1},a)$ l'estimation de la valeur future optimale. Enfin, $(r_t + \gamma \times \max_a Q(s_{t+1},a))$ est dite valeur apprise. De plus, lors de la phase d'entraînement, une valeur ε est établie permettant de définir le taux d'exploration. Pour implanter notre approche, nous avons utilisé la norme IEEE 802.11 en mode ad-hoc pour des communications entre drones, sans infrastructure ou point d'accès. Périodiquement, chaque drone diffuse un paquet contenant la liste des dernières mesures de RSSI (non discrétisées) et les sources associées. Les drones possèdent ainsi les valeurs RSSI de leurs voisins, permettant à chacun de calculer un état pour le Q-tableau.

3 Expérimentations

Nous avons implanté le protocole en utilisant ns-3 [RH10] et nous avons fourni les informations recueillies par l'échange des paquets contenant les RSSI à une application externe, à l'aide du système de publication-abonnement du système d'exploitation Robot (ROS) [Kou17]. Pour chaque drone, ns-3 propose une application qui stocke les valeurs des RSSI, les diffuse régulièrement (toutes les 200 ms) dans le simulateur et les met à disposition dans ROS. Un programme a également été développé afin de calculer une trajectoire cinématique pour le maître et les suiveurs. Celle du maître suit un chemin prédéterminé à une vitesse fixée, tandis que les suiveurs utilisent l'algorithme Q-learning pour maintenir la formation.Pour obtenir les informations RSSI, les suiveurs interrogent en permanence la partie ns-3 du système via le système de publication / abonnement, récupèrent les valeurs mises à jour, puis renvoient leur mouvement dans ns-3. Pour chaque expérimentation, nous avons imposé que les drones se déplacent en conservant une formation en triangle isocèle. La base du triangle mesure 200 m de large et 100 m de long. Le maître suit un chemin de 2000 m, comprenant trois virages à 90 degrés à 450, 950 et 1750 m. Les paquets sont diffusés à l'aide de la technologie 802.11n en mode ad-hoc, pouvant couvrir des centaines de mètres dans des conditions LOS. Pour les phases de formation et de tests, un modèle de propagation log-normal a été utilisé et la puissance de transmission a été réglé à 17 dBm. De plus, pour évaluer la réactivité des suiveurs par rapport aux mouvements du drone maître, nous avons fait varier la vitesse maximale de ce dernier tout en maintenant celle des suiveurs à 10 m/s. Nous avons également testé les performances de notre solution en fonction du nombre d'itérations pour la phase d'apprentissage de l'algorithme de Q-learning. Le résultat de notre analyse est l'erreur de positionnement, c'est-à-dire la distance euclidienne (à chaque instant) entre la position où les suiveurs devraient être pour conserver la formation initiale et leur position réelle.

Comme le montre la Figure 2, lorsque la phase d'apprentissage du Q-learning comprend au moins 2000 itérations (valeur obtenue par une campagne de simulation préliminaire), la formation est maintenue, l'erreur étant aux alentours de 1 ou 2 % par rapport aux 200 mètres séparant les drones. L'amplitude des oscillations est fonction de la vitesse maximale du drone maître (notée σ et exprimée en m/s), l'erreur étant naturellement plus faible lorsque celui-ci se déplace plus lentement. Ainsi, pour une vitesse maximale raisonnable, nous pouvons constater que l'erreur de positionnement est non seulement faible, mais qu'elle ne fluctue quasiment pas au cours du temps. Elle peut être réduite en considérant plus finement les puissances de signal, avec pour conséquence une augmentation du Q-tableau.

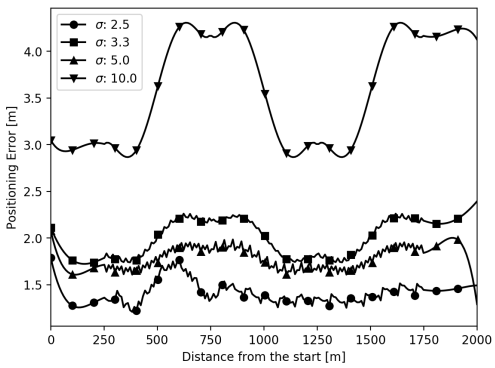

FIGURE 2: Erreur de positionnement

4 Conclusion

Nous avons proposé une solution efficace permettant à un opérateur, avec une seule télécommande, dans un enviromment extérieur ou intérieur, de contrôler un ensemble de drones se maintenant automatiquement en formation et offrant la qualité de service nécessaire à l'application en termes de débit. Notre approche s'appuie uniquement sur les valeurs des puissances de signal reçues lors des communications ad hoc au sein de la formation, et ne requiert donc aucune infrastructure ou matériel dédié. Les mouvements qu'un drone doit réaliser (hormis le maître piloté par l'opérateur) de manière autonome pour se (re)positionner correctement sont définis par un algorithme d'apprentissage par renforcement, le Q-learning. Notre solution a été implantée dans un protocole et testée avec ns-3. Les résultats sont concluants, montrant un taux d'erreur de positionnement stable et extrêmement faible.

Parmi nos perspectives, nous pouvons citer l'étude de l'effet des communications ad hoc multi-sauts sur la précision de la formation, mais également la détermination de la formation initiale au travers de la programmation robuste, en intégrant l'incertitude du positionnement et de la couverture des drones en fonction de l'application.

Références

[CBF18] Walton Pereira Coutinho, Maria Battarra, and Jörg Fliege. The unmanned aerial vehicle routing and trajectory optimisation problem, a taxonomic review. *Computers & Industrial Engineering*, 120 :116 – 128, 2018.

[CK02] Miclet L. Cornuéjols, A. and Y. Kodratoff. *Apprentissage Artificiel : Concepts et algorithmes*. Eyrolles, 2002.

[GTK17] Pradipta Ghosh, Jason A Tran, and Bhaskar Krishnamachari. Arrest : A rssi based approach for mobile sensing and tracking of a moving object. In *Globecom Workshops (GC Wkshps), 2017 IEEE*, pages 1–6. IEEE, 2017.

[HG17] S-M. Hung and S.N. Givigi. A q-learning approach to flocking with uavs in a stochastic environment. *IEEE Transactions Cybernetics*, 47(1) :186–197, 2017.

[KOK+10] T. Komatsu, T. Ohkubo, K. Kobayashi, K. Watanabe, and Y. Kurihara. A study of rssi-based formation control algorithm for multiple mobile robots. In *Proceedings of SICE Annual Conference 2010*, pages 1127–1130, Aug 2010.

[Kou17] Anis Koubâa. *Robot operating system (ros) : The complete reference*, volume 2. Springer, 2017.

[LB18] Yuanchang Liu and Richard Bucknall. A survey of formation control and motion planning of multiple unmanned vehicles. *Robotica*, 36(7) :1019–1047, 2018.

[Per08] Hogan J. Moulin B. Berger J. Bélanger M. Perron, J. A hybrid approach based on multi-agent geosimulation and reinforcement learning to solve a uav patrolling problem. In *WSC '08 Proceedings of the 40th Conference on Winter Simulation*, pages 1259–1267, 2008.

[RH10] George F Riley and Thomas R Henderson. The ns-3 network simulator. In *Modeling and tools for network simulation*, pages 15–34. Springer, 2010.

[Wat89] C.J.C.H. Watkins. *Learning from delayed rewards*. PhD thesis, University of Cambridge, 1989.

[WQXC14] Han Wu, Shizhen Qu, Dongdong Xu, and Chunlin Chen. Precise localization and formation control of swarm robots via wireless sensor networks. *Mathematical Problems in Engineering*, 2014, 2014.

[ZTHK17] W. Zhang, Y. Tang, T. Huang, and J. Kurths. Sampled-data consensus of linear multi-agent systems with packet losses. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11) :2516–2527, Nov 2017.